

無線 LAN における検索遅延削減を目的とした高速ハンドオフ

立野 哲也[†] 稲井 寛[†]

[†] 岡山県立大学情報工学部 〒719-1197 岡山県総社市窪木 111
E-mail: †{tateno,inai}@c.oka-pu.ac.jp

あらまし 無線 LAN では、移動端末がハンドオフを行っている間の通信不能な時間(遅延)が比較的大きくなるのが問題になっている。ハンドオフは検索、認証、接続の3つの手順によって構成されている。ハンドオフにおける遅延の大半は新たなアクセスポイントを検索するのに要する時間(検索遅延)であることが分かっている。そこで、本論文では、検索遅延の削減に焦点を当てる。隣接アクセスポイント間のハンドオフ関係を表す neighbor graph を用いることで、無駄なチャンネルの検索と無駄な待ち時間を削減する手法、キャリアセンスによってチャンネルの使用状況を推測し、無駄な待ち時間を削減する手法を提案した。シミュレーションによる性能評価の結果、検索遅延の削減が可能となることが示された。また、既存方式との組合せによって、更に検索遅延の削減が可能となることが示された。
キーワード 無線 LAN, ハンドオフ, 検索, Neighbor Graph, キャリアセンス

Fast handoff to reduce probe latency in wireless LANs

Tetsuya TATENO[†] and Hiroshi INAI[†]

[†] Faculty of Computer Science and System Engineering, Okayama, Prefectural University
111, Kuboki, Soja, Okayama, 719-1197 Japan
E-mail: †{tateno,inai}@c.oka-pu.ac.jp

Abstract In wireless LANs, it becomes a problem that the time which cannot communicate while performing the handoff becomes comparatively large. The handoff is constituted by three procedures, probe, authentication, and association. It is known that the most of handoff latency is the time which needs to probe new access points. For this reason, this paper focuses on to reduce the probe latency. We propose two new probe methods, the one reduces wasted probe wait and wasted channel by using neighbor graph which shows handoff relationship between access points, and the others reduces wasted probe wait by using carrier sense to speculate a channel state. Via simulation, the proposed methods reduces the probe latency. Moreover, it is shown that probe latency is reduced by combining existing methods and proposed methods.

Key words Wireless LANs, Handoff, Probe, Neighbor Graph, Carrier Sense

1. はじめに

IEEE 802.11 に準拠した無線 LAN が普及している。IEEE 802.11 は 2.4[GHz] で最大 11[Mbps] の IEEE 802.11b や、5[GHz] 帯で最大 54[Mbps] の IEEE 802.11a, 2.4[GHz] 帯で最大 54[Mbps] の IEEE 802.11g 等の規格を定めている。無線 LAN 技術は、室内環境におけるネットワーク機器間の接続をケーブルから解放するために設計された。よって、当初はハンドオフサポートは重要な問題であるとは考えられていなかった。しかし、近年の通信速度の向上に伴うマルチメディアアプリケーション等への適用において、アクセスポイント(以下「AP」と略記する)間でのハンドオフにおける遅延が重大な問題となってきた。

ハンドオフは検索、認証、接続の3つの手順で構成されている。端末は、まず、通信可能な AP の検索を行う。その後、発見した AP の中から1つを選び、その AP との間で認証を行う。認証が成功した後、接続を行う。ハンドオフを行っている間は、端末はハンドオフ以外の通信を行うことが出来ない。よって、マルチメディアアプリケーションや VoIP 等のリアルタイム通信において、この通信不能の時間が、通信品質の低下を引き起こす。VoIP において、通信不能な時間は最大でも 50[ms] であることが望ましいとされている [1]。しかし、ハンドオフにおける検索遅延は 60[ms] から 400[ms](平均 252[ms]) がかかることが分かっている [3]。また、ハンドオフにおける検索遅延の大半は、検索における検索遅延であることが分かっている [3]。これらの事より、本論文では、ハンドオフにおける検索遅延の削減

に焦点を当てる。

本論文に関連する研究として、隣接関係にある AP とのハンドオフ関係を表す neighbor graph [5] を用いる NG アルゴリズム [2]、隣接関係にある AP との非オーバーラップ関係を表す non-overlap graph [2] を用いる NG-Pruning アルゴリズム [2] が提案されている。これらの方式では、AP 数が増える毎に情報量が増加し、検索遅延が大きくなる。また、別の研究では、最初に応答を受信したチャンネルでの検索を終えた時点で検索を終了する Accelerated Probe Function [4] が提案されている。この方式では、検索が必要な全チャンネルで検索を行わないので、AP 数が増える毎に、その地点での最も電波強度の強い AP に接続する確率が低くなる。

上述の研究に対して、本研究では、neighbor graph を用いた場合での、検索遅延削減を行う方式の提案し、AP 数の増加による検索遅延、最も電波強度の強い AP に接続する確率について、上述の方式と比較する。

以下、2. では、IEEE 802.11 にて規定されている検索方法について述べる。3. では、上述した既存方式の概要を説明する。4. では、提案方式の説明を行う。5. では、シミュレーションモデルの説明を行う。6. では、シミュレーション結果を示し、性能評価を行う。最後に、7. では、本論文で得られた結果のまとめと今後の課題について述べる。

2. 標準方式

移動端末は、ハンドオフを行う際に、新しく通信を行う AP を探す必要がある。移動端末は、AP とプローブ要求 (Probe Request)/プローブ応答 (Probe Response) フレームの交換によって周辺に存在する AP の検索を行う。図 1 に検索方法の概略図を示す。移動端末は検索を行うチャンネルを決定し、プローブ要求をブロードキャストする。AP はプローブ要求を受信すると、プローブ要求送信元へプローブ応答を送信する。移動端末は、プローブ要求送信後、タイマを起動して、応答を受信するために、チャンネルを監視する。チャンネルの監視時間には MinChannelTime (最小チャンネル監視時間) と MaxChannelTime (最大チャンネル監視時間) の 2 つがある。移動端末は、タイマが MinChannelTime に達した時点で AP からの応答が無かった場合、そのチャンネルには AP は存在しないものと解釈して、次のチャンネルで検索を開始する。MinChannelTime 以内に応答があった場合、移動端末は ACK を返し、チャンネルの監視を続ける。その後、MaxChannelTime 以内に応答があった場合は同様の動作を行う。タイマが MaxChannelTime に達した時点で、そのチャンネルでの検索を終了して、次のチャンネルで検索を開始する。以上の動作を全てのチャンネルについて行う。

標準方式では、全チャンネルで検索を行うので、不要なチャンネル (Wasted Channel) での検索が多く発生する。また、MinChannelTime 以内に応答があった時に、MaxChannelTime まで応答を待つが、そのチャンネルでの AP が少ない環境では不要な待ち時間 (Wasted Probe-Wait) が発生する。よってこれらの不要な時間を削減することが、検索遅延の削減につながる。

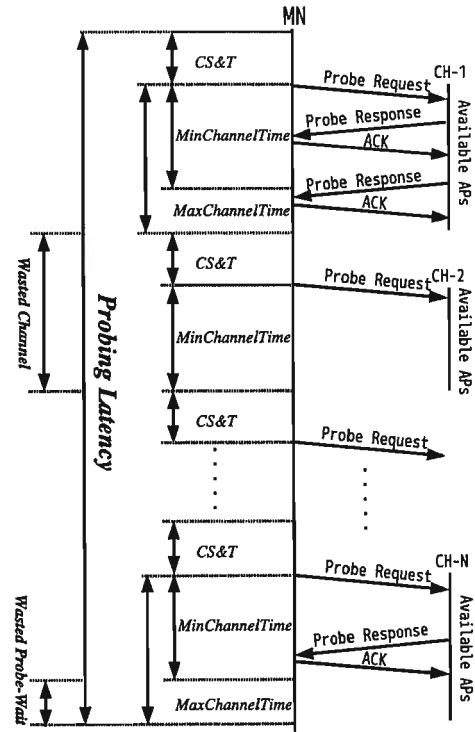


図 1 検索方法

3. 既存方式

3.1 NG アルゴリズム

3.1.1 Neighbor Graph

neighbor graph (NG) [5] は隣接 AP 間のハンドオフ関係を表す。ハンドオフ関係とは、ある AP から隣接関係にある別の AP へハンドオフが可能であるという関係を表す。例えば、 AP_x が AP_y に対してハンドオフ関係があるとすると、neighbor graph は (AP_x, AP_y) という情報を持つ。ただし、 AP_y から AP_x へハンドオフ関係があるかどうかは分からないので、 $(AP_x, AP_y) \cup (AP_y, AP_x)$ である。このような情報を持つ neighbor graph の生成方法は 2 つある。1 つめは、静的に neighbor graph を生成する方法である。最初に生成し、その後はその情報を変化させないようにする。しかし、この方法では、AP の移動、障害物等によって、AP 間のハンドオフ関係が変わった時に対応出来ない。2 つめは、動的に neighbor graph を生成する方法である。例えば、 AP_i から AP_j へハンドオフし、neighbor graph にこの情報 (AP_i, AP_j) が存在しないとき、この情報を追加する。この方法なら、AP 間のハンドオフ関係が変化しても、1 度ハンドオフを発生させ情報を追加することで、変化に対応できる。

3.1.2 NG アルゴリズムの概要

NG アルゴリズムは neighbor graph を用いるハンドオフ手

法である。NG アルゴリズムでは、まずハンドオフの際に現在サービスを受けている AP をキーとする neighbor graph を調べ、いずれかの隣接 AP が動作しているチャンネルでプローブ要求をブロードキャストする。プローブ要求送信後、タイマを起動して、応答を受信するためにチャンネルを監視する。タイマが MinChannelTime に達した時点で応答が 1 つも無ければ、neighbor graph を調べ、別のチャンネルで検索を行う。一方、neighbor graph に存在する現在検索を行っているチャンネルで動作している全ての隣接 AP から応答があった場合も、別のチャンネルで検索を行う。あるいは、タイマの値が MaxChannelTime に達したら、別のチャンネルで検索を行う。これらの動作を全隣接 AP が動作しているチャンネルで行う。

この方式では、初期の段階では情報量が少なく、十分な効果を期待できないが、ハンドオフ回数を重ねることで、十分な効果を期待できる。ただし、ハンドオフ回数は端末によって異なるので、neighbor graph の情報をネットワーク内で共有できる環境を作ることによって、この方式が効果を発揮できるようになる。この環境を作るためには、ネットワーク内の neighbor graph の情報を管理するサーバーを用意するか、AP 間で neighbor graph の情報を共有し、接続を行った端末に、自身の持つ情報を送信する等が考えられる。

3.2 NG-Pruning アルゴリズム

3.2.1 Overlap Graph, Non-Overlap Graph

overlap graph(OG) [2] とは、AP 同士のサービスエリアがオーバーラップしている AP の集合を表す。{ AP_i, AP_j } は、 AP_i と AP_j がオーバーラップの関係にある事を表している。つまり、サービスエリアがオーバーラップしている位置でハンドオフが発生すれば、 AP_i, AP_j のどちらとも接続する可能性がある。

non-overlap graph(NOG) は AP 同士のサービスエリアがオーバーラップしていない AP の集合を表す。

3.2.2 NG-Pruning アルゴリズムの概要

NG-Pruning アルゴリズムは、overlap graph と non-overlap graph を用いるハンドオフ手法である。NG-Pruning アルゴリズムでは、現在サービスを受けている AP の overlap graph に含まれる AP の non-overlap graph を調べ、非オーバーラップの度合いが最も高い、つまり、オーバーラップの度合いが最も低い AP が動作しているチャンネルで検索を行う。プローブ要求をブロードキャストし、タイマを起動後、チャンネルの監視を行う。タイマの値が MinChannelTime に達しても応答が得られなかったときは、上記の方法で別のチャンネルで検索を行う。応答があった場合、まず応答を行った AP の non-overlap graph を調べ、その non-overlap graph に含まれる AP を検索対象から除外する。その後、現在検索を行っているチャンネルで動作する全ての隣接 AP が応答を行うか、検索対象から除外された場合、別のチャンネルで検索を行う。それ以外の場合は、MaxChannelTime まで応答を待ち、その後、別のチャンネルで検索を行う。これらの動作を、全ての隣接 AP が応答を行うか、検索対象から除外されるまで行う。

この方式は、NG アルゴリズムと同様に、ハンドオフの経験

によってネットワークの状態を知る事で、十分な効果を発揮することが出来る。そのためには、NG アルゴリズムと同様に、overlap graph, non-overlap graph をネットワーク内で共有できる環境を作ることによって、この方式が効果を発揮できる。

3.3 Accelerated Probe Function

NG アルゴリズム、NG-Pruning アルゴリズムはハンドオフの経験により、ネットワークの状態を知ることでハンドオフによる検索遅延の削減を行っている。しかし、ネットワークの状態を知るための情報が十分になるまでには、ある程度ハンドオフを発生させなければならず、情報が十分にならない内はこれらの方式では十分な効果を期待することは出来ない。そこで考えられたのが、Accelerated Probe Function [4] である。この方式において、必要な情報は現在接続している AP との通信に用いているチャンネルの情報だけなので、ハンドオフを発生させ、情報を増やす必要が無い。Accelerated Probe Function での検索方法は、ハンドオフが発生したら、まず、現在接続している AP との通信に用いているチャンネルを確認する。その後、そのチャンネルの次のチャンネルから検索を開始する。あるチャンネルでの検索の方法は標準方式と同様である。標準方式との違いは、標準方式が全チャンネルで検索するのに対して、Accelerated Probe Function では、最初に応答があったチャンネルまでしか検索を行わないことである。それによって、検索遅延の削減を行うことが出来るが、最適な AP との接続確率は他の方式よりも低くなる事が想像できる。

4. 提案方式

4.1 提案方式 1

neighbor graph と overlap graph の違いは、有向であるか無向であるか。また、NG アルゴリズムと NG-Pruning アルゴリズムの違いは、neighbor graph を用いるか non-overlap graph を用いるかという事と、応答のあった AP の情報を基に、検索不要な AP を知ることが出来るか否かという事である。まず、neighbor graph は単方向の有向グラフであるが、ハンドオフを発生させ、情報の変化が無くなったとき、neighbor graph が単方向であるか双方向、つまり無向であるかどうかを調べた。その結果が表 1 である。

表 1 AP1 台当たりの Neighbor Graph の単方向グラフ数

AP 数	単方向グラフ数
3	4.390000e-03
5	4.163800e-02

また、neighbor graph の情報量と、overlap graph の情報量の比較を行ったのが表 2 である。表 2 を見ると、neighbor graph は overlap graph にある程度近い情報量を持っている事が分かる。

これらの事から、neighbor graph を用いて、NG-Pruning アルゴリズムの様に、応答のあった AP の情報を用いることで、検索における検索遅延の削減が可能となると考えた。

チャンネルの選択方法について、NG アルゴリズムでは、隣接

表 2 AP1 台当たりの overlap graph と neighbor graph の情報量の差

AP 数	情報量の差
3	2.432333e-02
5	2.082940e-01

AP が動作するチャンネルからランダムに選んでいたが、提案方式では、neighbor graph の使用回数を記憶し、現在接続している AP をキーとする neighbor graph での使用回数の多い順に、neighbor graph に含まれる隣接 AP の動作しているチャンネルで検索を行う。

提案方式では、移動端末はハンドオフの際に、neighbor graph を調べ、上述の方法で検索チャンネルを決定し、検索を行う。プローブ要求をブロードキャストした後、チャンネルの監視を始める。MinChannelTime 以内に応答が無ければ、別のチャンネルで検索を開始する。応答があった場合、応答を行った AP をキーとする neighbor graph を調べ、その neighbor graph 内に含まれない AP を検索対象から除外する。その後、現在検索を行っているチャンネルで動作する他の隣接 AP が応答を行うか、検索対象から除外されたら、別のチャンネルで検索を行う。それ以外の場合は、MaxChannelTime までチャンネルの監視を続け、応答を待つ。これらの動作を検索の必要のある全チャンネルで行う。

この方式では、overlap graph よりも情報の少ない neighbor graph を用いるので、必要な情報を除外してしまう可能性があるが、その分、NG-Pruning アルゴリズムよりも早い時間で検索が行えることが推測できる。

4.2 キャリアセンス

IEEE 802.11 無線 LAN で通信を行う端末は、キャリアセンスを行うことでチャンネルの使用状況を知ることが出来る。通信量が多い環境では、頻りにキャリアを検知する。逆に、通信量が少ない環境では、あまりキャリアを検知しない。また、送信したいデータを持っている状態では、通信量が多い環境では、他の端末が先に通信を始める可能性が高く、なかなか通信を行うことが出来ない。通信量が少ない環境では、ほとんど他の端末に邪魔されること無く、通信を開始出来る。

標準方式での検索を考えると、通信量が少ない環境では、プローブ要求を受信した AP は、バックオフ時間のみを待つすぐにプローブ応答を送信する可能性が高い。通信量が多い環境では、プローブ応答を送信するまでに、他の端末が邪魔をする可能性が高い。

これらの事より、移動端末は、プローブ要求送信後に一定時間キャリアセンスを行い、キャリア検知が出来なかった場合、そのチャンネルは誰も使用していないと考え、その時点でそのチャンネルでの検索を終了する。この考え方を、各方式に適用することで、不要な待ち時間の削減が可能となる考えられる。

4.3 提案方式 2

3.3 で述べた Accelerated Probe Function について考える。この方式では最初に応答のあったチャンネルで検索を終了する。AP 数が少ない時はこの方式は有効であると考えられるが、AP 数が増えると、最適な AP と通信する事ができず、通信サービ

スの低下、ハンドオフ発生回数の増加等の悪影響が発生する。そこで、この方式に対して、検索の終了条件に冗長性を持たせる事で、AP 選択に幅を持たせる事を考える。また、4.2 の考え方を用いる事で、検索遅延の削減を行う。検索の終了条件を以下の様にする。

- (1) 2つ以上の AP から応答を得た時
 - (2) 1つの AP から応答を得て、2つ以上のチャンネルでキャリアを検知した時。
 - (3) 周波数が独立しているチャンネルをグループ化する (IEEE 802.11b の場合、{(1,6,11,14), (2,7,12), (3,8,13), (4,9), (5, 10)} の5つのグループ)。あるグループ内のチャンネルで応答を得ている時に、そのグループ内の全チャンネルでの検索を終了した時。
- これらの終了条件を用いる事で、Accelerated Probe Function よりも最適な AP へ接続する確率を上げることが出来る。

5. シミュレーションモデル

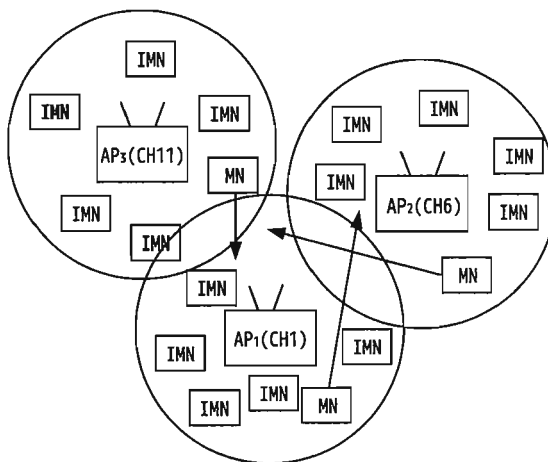


図 2 シミュレーションモデル

シミュレーションモデルを図 2 に示す。MN は移動端末 (Mobile node), IMN は固定端末 (Immobile node) を表している。以下にシミュレーションモデルの要件を示す。

- (1) 各 AP のサービスエリアは各 AP を中心とする円とする。サービスエリア半径は 60[m] とする。
- (2) 中心に AP (AP_c とする) を配置し、隣接 AP を AP_c の周りにランダムに配置する。隣接 AP 台数を $m(\{AP_1, AP_2, \dots, AP_m\})$, AP のサービスエリアの半径を R とした時、AP の配置方法は以下に従う。

$$\left\{ \begin{array}{l} R < \text{Distance}(AP_i, AP_c) < 2 \times R \\ \text{ただし, } i = 1, 2, \dots, m \\ \\ R < \text{Distance}(AP_i, AP_j) \\ \text{ただし, } j = 1, 2, \dots, m; i \neq j \end{array} \right.$$

ここで、Distance() は、AP 間の距離を表す。

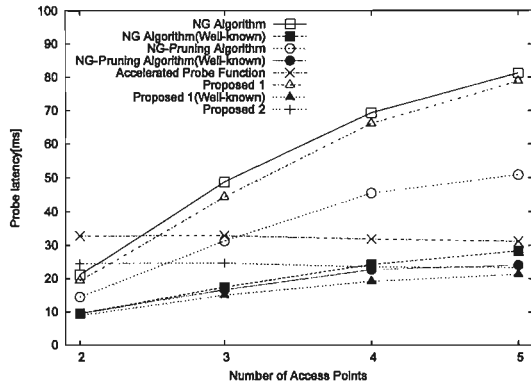


図3 検索遅延の比較

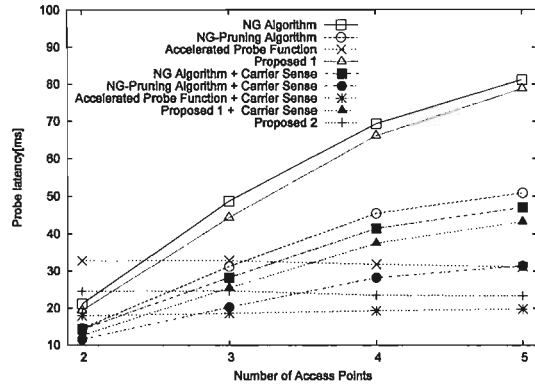


図4 キャリアセンスが検索遅延に及ぼす影響

(3) AP 同士がオーバーラップの関係にある時、AP 間の距離は

$$Distance(AP_i, AP_j) \leq 2 \times R$$

(4) 同一チャンネルで動作する AP 同士は、

$$Distance(AP_i, AP_j) \geq 2 \times R$$

となる。

(5) 端末は、固定端末と移動端末に分けられ、固定端末は、移動を行わず、AP との通信のみを行う。移動端末は、移動を行いながら AP と通信を行い、ハンドオフも行う。

(6) 発生するデータは、ペイロードが 208[Byte] で平均 20[ms] の間隔で発生する。これは G.711-coded VoIP に相当する。

(7) 移動端末の移動方向は、サービスを受けていた AP からサービスが受けられない位置に移動しようとした時、最低 1 台以上の別の AP のサービスエリア内に居るような方向をランダムに選択する。

(8) 端末が AP からサービスを受けられるのは、

$$Distance(MN, AP_i) \leq R$$

の時だけである。

これらの要件を満たして、シミュレーションを行う。全移動端末は平均 20 回のハンドオフを行う。シミュレーションに用いたパラメータを表 3 に示す。尚、本研究では、無線 LAN 規格として IEEE 802.11b を用いている。

表 3 シミュレーションに用いたパラメータ

MaxChannelTime	11[ms]
MinChannelTime	7[ms]
Channel Switch	5[ms]
Number of immobile nodes per AP	5
Number of mobile nodes per AP	1

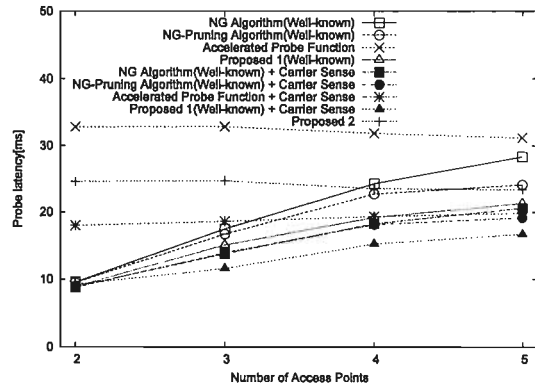


図5 キャリアセンスが検索遅延に及ぼす影響(隣接情報が既知の場合)

6. 性能評価

図 3～図 5 は、配置する AP 数を変化させたときの各方式毎の検索遅延、図 6～図 8 は、配置する AP 数を変化させたときの各方式毎の最適な AP から応答を得た確率を表している。尚、本研究では、電波強度最大の AP を最適 AP とし、電波強度は距離によってのみ変化するものとする。つまり、ハンドオフが発生した地点から最も近い AP を最適 AP とみなす。図 3, 5, 6, 8 の Well-known とは、neighbor graph, overlap graph が既知である事を指している。逆に、何も記されていない場合は、neighbor graph, overlap graph は空の状態からシミュレーションを開始している。また、全ての方式において、1つの AP から応答が得られなかった場合、標準方式を用いて検索を行う。

まず、検索遅延について比較を行う。図 3 を見ると、情報が空の状態から開始した場合、提案方式 1 は、NG アルゴリズムより僅かに検索遅延が小さいだけだが、既知の状態では、どの既存方式よりも検索遅延が小さくなっている。提案方式 2 は、Accelerated Probe Function よりも僅かに検索遅延が小さくなっている。AP 数が 5 台の時に、NG アルゴリズムと同程度の示している。NG アルゴリズム、NG-Pruning アルゴリズム、提案方式 1 は、AP 数が増える毎に検索遅延が大きくなって

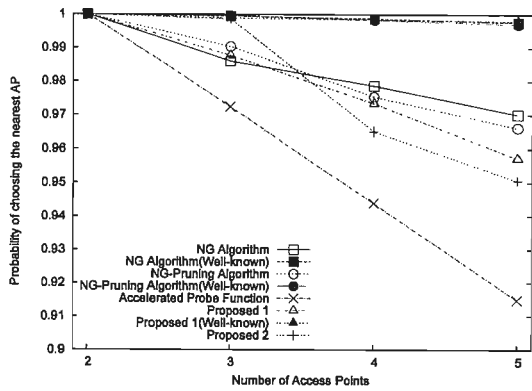


図6 最適 AP 選択確率の比較

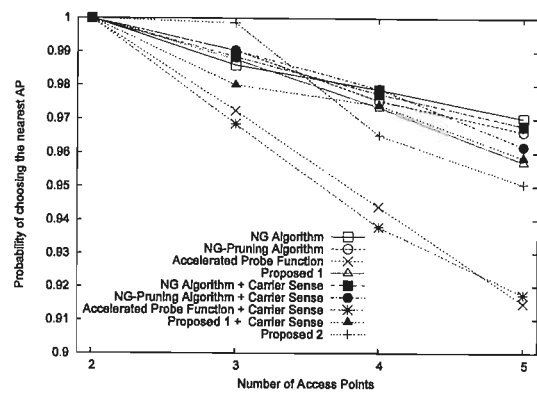


図7 キャリアセンスが最適 AP 選択確率に及ぼす影響

る。これは、neighbor graph, overlap graph が、AP 数が増えるたびに情報量が増えるためである。図 4, 5 では、各方式に対して、4.2 で述べたキャリアセンスの考え方を適用している。尚、キャリアを 1[ms] 以内に検知出来なかった場合に、検索チャンネルを移行することにした。適用前と比べると、全ての方式で検索遅延が小さくなっていることが分かる。情報が既知の状態では、提案方式 1 に適用した場合が、最も検索遅延が小さくなっている。

次に、最適 AP 選択確率について比較を行う。図 6 を見ると、Accelerated Probe Function が最も確率が低く、情報が既知の場合、NG アルゴリズム、NG-Pruning アルゴリズム、提案方式 1 がほぼ同様の結果を示している。空の状態から開始した場合は、提案方式 1 は、NG アルゴリズム、NG-Pruning アルゴリズムよりも低い値を示している事が分かる。また、これらの方式では、情報が既知の時では、最適 AP 選択確率が 99% 以上の値を示している。提案方式 2 は、Accelerated Probe Function より高い値を示しているが、その他の方式と比べると低い値を示している。この原因は、応答を行う AP 数が増えるため、また、AP 数が増加する毎に、固定端末が増加し、移動端末が接続出来ない AP と通信をする端末のキャリアを検知してしまうためである。図 7, 8 を見ると、キャリアセンスの考え方を適用しても、最適 AP 選択確率が大きく変化することは無いことが分かる。

これらの事より、提案方式 1 では、情報が既知であれば、検索遅延が削減される。提案方式 2 では、Accelerated Probe Function に比べ、検索遅延が削減され、最適 AP 選択確率も改善される。また、キャリアセンスを用いることで、最適 AP 選択確率を低下させずに検索遅延を削減をすることが可能となる。

7. おわりに

本稿では、ハンドオフの AP 検索時において、neighbor graph を用いたハンドオフ方式、キャリアセンスによるチャンネル切替のタイミングについて提案した。neighbor graph の情報が既知である時には提案方式 1 が最も小さい検索遅延を示し、また、キャリアセンスの考え方を適用することで、全ての方式で

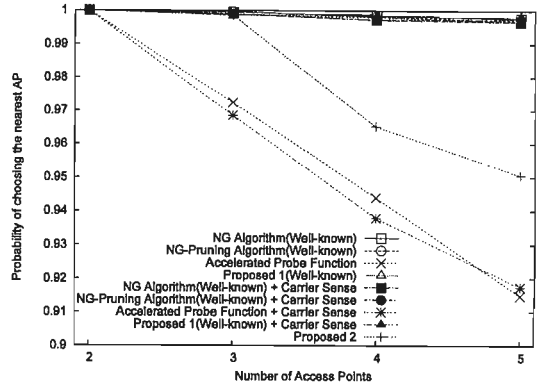


図8 キャリアセンスが最適 AP 選択確率に及ぼす影響 (隣接情報が既知の場合)

検索遅延の削減が可能となることを確認した。

今後の課題としては、neighbor graph 等の情報を、ネットワークで共有する手法について考える必要がある。また、その際にどのような影響が出るかを調べる予定である。

文献

- [1] International Telecommunication Union, "General Characteristics of International Telephone Connections and International Telephone Circuits.", ITU-TG.114, 1988
- [2] M. Shin, A. Mishra, and W. Arbaugh, "Improving the Latency of 802.11 Handoffs using Neighbor Graphs", Proc. ACM Mobisys 2004, June 2004.
- [3] A. Mishra, M. Shin, and W. Arbaugh, "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process", ACM SIGCOMM Computer Communications Review, vol. 33, no. 2, Apr. 2003.
- [4] L. J. Zhang, S. Pierre, and L. Zhang, "Fast Handoff Scheme with Accelerated Probe Function in Wireless LANs", Proc. of the Seventh Lasted International Conferences Wireless and Optical Communications, May 30-June 1 2007.
- [5] A. Mishra, M. Shin, and W. A. Arbaugh, "Context Caching using Neighbor Graphs for Fast Handoffs in a Wireless Network", Proc. IEEE INFOCOM 2004, Mar. 2004